

AMENDMENTS TO THE CLAIMS

1. (Currently Amended) A method; comprising:

monitoring a node associated with a contended lock; ~~and~~
identifying a processor waiting for the contended lock;
~~putting a~~the ~~processor that is waiting for the contended lock to become available~~
to sleep until an event occurs;
relinquishing resources of the sleeping processor; and
forming larger resources for non-sleeping processors to utilize while the processor
sleeps, the larger resources including the relinquished resources.
2. (Currently Amended) The method of claim 1, wherein the monitoring of the node comprises monitoring a lock address corresponding to the contended lock by executing a monitor instruction to activate the monitoring of the node.
3. (Currently Amended) The method of claim 1, further comprises executing a memory wait an (mwait) instruction to put the processor to sleep until the event occurs.
4. (Original) The method of claim 1, further comprises:

waking up the processor when the event occurs, wherein the event comprises the contended lock becoming available; and
the processor acquiring the available lock.
5. (Currently Amended) The method of claim 1, wherein the ~~contended lock~~
~~becoming available comprises the~~ processor is next in a queue to acquire the contended lock, ~~and the contended lock is released.~~
6. (Cancelled)

7. (Currently Amended) The method of claim 4, wherein the waking up of the processor comprises inactivating the monitoring of the node, and the processor using reacquiring the relinquished resources for the awakened processor to utilize.
8. (Currently Amended) The method of claim 6_1, wherein the relinquishing of the resources comprises:
 - relinquishing of a plurality of registers in a register pool;
 - relinquishing of a plurality of instruction queue entries in an instruction queue;
 - relinquishing of a plurality of store buffer entries in a store buffer; and
 - relinquishing of a plurality of re-order buffer entries in a re-order buffer.

Claims 9-12 (Cancelled)

13. (Currently Amended) A processor, comprising:

a monitor an execution unit to execute a monitor instruction and a memory wait (mwait) instruction to

monitor a node associated with a contended lock, and

identify a processor waiting for the contended lock; and

logic to put a logical the processor that is waiting for the contended lock to

become available to sleep until an event has occurred; and

a resource manager to

relinquish resources of the sleeping processor,

form larger resources for non-sleeping processors to utilize while the

processor sleeps, the larger resources including the relinquished

resources.
14. (Original) The processor of claim 13, further comprising detection logic to detect the occurrence of the event, wherein the event comprises a designated event

- including the contended lock becoming available.
15. (Cancelled)
16. (Currently Amended) The processor of claim 13, wherein the logic is further to wake up the logical processor when the event occurs, the waking up comprises inactivating the monitoring of the node, and reacquiring the logical processing using relinquished resources for the awakened processor to utilize.
17. (Currently Amended) The processor of claim-~~15~~ 13, wherein the relinquishing comprises resource manager is further to:
- ~~relinquishing of relinquish~~ a plurality of registers in a register pool;
- ~~relinquishing of relinquish~~ a plurality of instruction queue entries in an instruction queue;
- ~~relinquishing of relinquish~~ a plurality of store buffer entries in a store buffer; and
- ~~relinquishing of relinquish~~ a plurality of re-order buffer entries in a re-order buffer.
18. (Currently Amended) A system comprising:
- a storage medium; and
- a processor coupled with the storage medium, the processor having
- a monitor an execution unit to execute a monitor instruction and a memory mwait (mwait) instruction to
- monitor a node associated with a contended lock, and
- identify a processor waiting for the contended lock;
- logic to put a logical ~~the~~ processor that is waiting for the contended lock to become available to sleep until an event has occurred; and
- a resource manager to

relinquish resources of the sleeping processor,
form larger resources for non-sleeping processors to utilize while the
processor sleeps, the larger resources including the relinquished
resources.

19. (Original) The system of claim 18, further comprising detection logic to detect the occurrence of the event, wherein the event comprises a designated event including the contended lock becoming available.
20. (Cancelled)
21. (Currently Amended) The system of claim 18, wherein the logic is further to wake up the logical processor when the event occurs, the waking up comprises inactivating the monitoring of the node, and reacquiring the logical processing using relinquished resources for the awakened processor to utilize.
22. (Currently Amended) A machine-readable medium having ~~stored thereon data representing sets of instructions, the sets of instructions which, when executed by a machine, cause the machine to:~~ monitor a node associated with a contended lock; ~~and identify a processor waiting for the contended lock;~~ put ~~a~~~~the processor that is waiting for the contended lock to become available to~~ sleep until an event occurs;
relinquish resources of the sleeping processor; and
form larger resources for non-sleeping processors to utilize while the processor
sleeps, the larger resources including the relinquished resources.
23. (Currently Amended) The machine-readable medium of claim 22, wherein the instructions when executed to monitoring monitor the node ~~comprises monitoring~~

to further cause the machine to monitor a lock address corresponding to the contended lock by executing a monitor instruction to activate the monitoring of the node.

24. (Cancelled)
25. (Currently Amended) The machine-readable medium of claim 22, wherein the sets of instructions which, when further executed by the machine, further cause the machine to:
wake up the processor when the event occurs, wherein the event comprises the contended lock becoming available; and
allow the processor to acquire the available lock.

Claims 26-30 (Cancelled)